Onderzoeksrapport Nr. 7803

ON THE MEASUREMENT OF COMPLEXITY IN ACTIVITY NETWORKS

by

Salah E. ELMAGHRABY[+]
Willy HERROELEN[++]

+ North Carolina State University at Raleigh, N.C.
++ The Catholic University of Leuven, Belgium.

This research was partially supported by Grant No. DAAG29-76-G-0204 with the U.S. Army Research Office.

Wettelijk Depot Nr. D/1978/2376/04.

ABSTRACT

The measurement of the "complexity" of activity networks seems to be needed in order to estimate the computing requirements and/or to validly compare alternative heuristic procedures. This paper critically evaluates past contributions to the problem, and explores the underlying concepts of measurement. It suggests that the objective of analysis of the network is a determining factor in the process of measurement, and discusses three different objectives; they are: to determine the critical path assuming deterministic time estimates; to determine the probability distribution function of project completion assuming random durations of the activities; and to determine the optimal schedule under limited availability of a single resource. It proposes a measure of network complexity for each objective.

1. INTRODUCTION

Measures of network "complexity" within the context of Activity Networks (ANs) have been receiving attention from researchers since the mid-sixties. This paper discusses the fundamental precepts (and inadequacies) of such "measures", presents an expository treatment of the basic concepts of the theory of measurement, and offers three "measures of network complexity" as illustrations of the propositions outlined in our discussion. The two appendices at the end of the paper present, respectively, a more detailed review of the past contributions of which we are aware, and the proofs of the formulae used in counting feasible sequences in Section 3.

The introduction presents three preliminaries to the main body of the paper; to wit:

(i) The need for measures of complexity of ANs;

(ii) The relation of such measures to combinatorial complexity theory;

(iii) Resumé of current approaches.

(i) The Need for Measures of Complexity

Whenever the words "network complexity" are mentioned they immediately evoke the question: What is meant by "network complexity", and why measure it?

To answer, we remind the reader that all measurement starts with a sensation, and we suggest that the sensation felt here is that of difficulty in analysis and synthesis of a given network. Henceforth we take this as our definition of "network complexity". That this difficulty is "real" - in the sense of it being felt by a large majority of workers in the field - is an empirically verifiable assertion.

The physical sciences are replete with examples illustrating the origin of measurement in a sensation: one feels cold or hot and one inquires about the measurement of "heat"; or one feels that a substance is more amenable to elongation than another and one inquires about its "elasticity"; or one feels that a fluid is more sticky than another and one inquires about its "viscosity"; and so on. Indeed, modern technology attests to human triumph in according measures to these sensations and, more significantly, in manipulating the factors that determine their magnitudes.

Therefore, there is no doubt in our minds about either the reality of the sensation of "complexity" of ANs, or the need for its measurement. We, among others, accept the premise that questions of measurement lie at the very foundation of scientific progress, and that the field of study of ANs is no exception.

Turning now to the "why" - which we interpret as the usefulness of such a measure - we propose (with Kaimann [23]) two possible applications:

(i) The measure would serve as a predictor of the processing time requirements for a particular software package for a particular piece of hardware;

(ii) The measure is necessary for comparing two (or more) proposed algorithms for the analysis of ANs.

Evidently, a choice between two proposed algorithms, or the determination of the efficiency of a particular algorithm, would be greatly facilitated if there exists a measure of network complexity. This would eliminate any possible bias in the conclusions regarding the efficacy of a particular algorithm relative to others by ensuring that the algorithm is evaluated at several points in the "range of complexity".

(ii) Relation to Combinatorial Complexity Theory

It seems also appropriate to clarify, at the outset, another issue that may have been suggested to the reader by the title of the paper, namely, the relationship, if any, between the "measures of network complexity" (MNC) discussed here, and the modern "theory of combinatorial complexity" as expounded by Edmonds, Cook, Karp, Lawler, Lenstra, and Rinnooy Kan (for a succinct and clear exposition of this theory, see refs. [2,7,37, 47]).†

At the risk of oversimplifying what is evidently a more involved theory than we can hope to expound in a few sentences, one may summarize that theory in the following manner. Combinatorial problems, such as the optimal scheduling of tasks on processors, fall into one of two categories: those problems that are solvable by an algorithm in time that is of polynomial order in the size of the problem (abbreviated: "in polynomial time"), and those problems that are not solvable by an algorithm in polynomial time. Here, "polynomial time" refers to the highest order of the polynomial that best fits the time data obtained empirically (or derived analytically from the count of "elementary operations"). The "size of the problem" is precisely defined in terms of the binary input to a single-tape Turing machine -- but we need not concern ourselves with these technical details here. Suffice it to say that the first class is called the P-class of problems, and the second the NP-class. A

---

†We take it for granted that the reader realizes that there is no relation whatsoever between our concern in this study and the mathematical theory of measure.

problem falling in the class NP is called an "NP-hard" problem, and the growth of time to solve such problems is of exponential (or higher) order. In the last couple of years the dichotomy between the P and NP classes has been enriched by the addition of a third class, referred to as the "pseudo-NP' class. But, again, we need not delve into the manner of classification into one class or another, since that would take us far afield from our primary objectives.

For our purposes, it is sufficient to remark that the classification of a problem as in the class P merely describes the worst-case performance of the computing effort, and does not necessarily reflect the actual computing effort required to solve a particular problem. Moreover, two algorithms are judged to be "equivalent" if they solve the same class of problems in time of the same polynomial order, say both are of $O(n^2)$, where n is the number of individual objects in the original statement of the problem. This is evidently a min-max criterion and, typically, it ignores the measure of effort (which we take to be equivalent to difficulty or complexity) except in the worst case.

On the other hand, the classification of a problem as in the class NP is basically a statement on the asymptotic behavior of the class of problems under discussion. Thus, e.g., the classification of the traveling salesman problem (TSP) as NP-hard more appropriately describes the manner of growth of the computing effort with the number of cities, rather than be a measure (i.e., estimate of magnitude) of the time it takes to solve a TSP of, say, 15 cities on an IBM 360/75.

We therefore conclude that we are concerned in this study with a different measure of complexity than that offered by the theory of combinatorial combinatorial complexity, since we are interested in isolating the factors that

determine the computing effort, and in achieving a calibration of the scale that characterizes such effort.

(iii) Overview of Current Approaches

Contributions to measures of network complexity emanate from studies of ANs as well as from two closely allied areas: Assembly Line Balancing (ALB) and Machine Scheduling problems. Table 1 gives a bird's eye view of the proposed measures; a more detailed discussion of the various approaches is given in Appendix A. We hasten to add that the three reports of Davis, Patterson, and Gordon, grouped under "Experiments" in Column 1, address the problem of network complexity only obliquely, since their primary objective was to estimate the relative contribution of different factors to the percent elongation in project duration (beyond the (unconstrained) length of the critical path) caused by the scarcity of resources.

The suggested measures of Pascoe, Davies and Kaimann rely totally on the count of the activities and nodes in the network. Since it is easy to construct networks of equal number of arcs and nodes but varying degrees of difficulty in analysis, (see Appendix A), we fail to see how these "measures" can discriminate among them!

Both Ignall and Thesen are concerned with the count of sequences that can be generated from a given AN. Note the important qualitative difference between the two problems addressed by these two authors. Since Ignall is treating the ALB problem, a translation of the assignment-of-tasks-to-work stations, which is the format in which the ALB problem is stated, into the format of ANs implies that each activity is of duration one and "consumes" an amount of the resource equal to the given duration of the originally-defined task of the ALB.

ACTIVITY NETWORKS	LINE BALANCING	MACHINE SCHEDULING
-Coefficient of Network Complexity CNC-P = A/N Pascoe (1966) CNC-D = $2(A-N+1)/(N-1)(N-2)$ Davies (1973) CNC-K = A^2/N Kaimann (1974) -Total Activity Density-T-Density $\sum_{N}$ max{0; # of predecessor activities - # of successor activities) Johnson (1967) -Average Activity Density $\frac{\text{T-Density}}{N}$ Patterson (1976) -Restrictiveness $R = 1 - \log S / \log S_{max}$ Thesen (1975) S = number of feasible sequences -Order Strength Cooper (1976) borrowed from line balancing -Experiments Davis (1975) Patterson (1976) Gordon (1976)	-Feasible Subsets Held, Karp & Shareshian (1963) -Feasible Sequences Ignall (1965) $\frac{n!}{2^r}$ r = No. of precedence relations -Order Strength $\frac{\text{\# ordered pairs}}{N(N-1)/2}$ Mastor (1970) -Flexibility Ratio $\left(\frac{\text{\# zeros in half matrix}}{N(N-1)/2}\right)$ Dar-El (1973)	-Dependent Shop Spencer (1969) $MAXPOS(DS) = \prod_{i=1}^{m} \frac{n(j)!}{DENOM(j)}$ j = 1,2,...,m facilities where SET(j) = nodes requiring facility j n(j) = total # activities requiring facility j $DENOM(j) = \prod_{i=1}^{n(j)}$ [1 + # activities requiring facility j which are dominated by node i, where i,j $\in$ SET(j)]

Table 1. Summary of Measures of Network Complexity

There is only one resource of availability equal to the given cycle time of the original ALB problem. On the other hand, Thesen assumes no such restriction on either the number of resources or their availabilities, in addition to the arbitrary durations of the activities.

Ignall's count of $n!/2^r$ is evidently an upper bound on the desired count, and is therefore ineffectual as a measure of complexity except in extreme cases. On the other hand, Thesen's measure of restrictiveness, while eminently plausible, is non-verifiable because of the impossibility of evaluating S, the number of feasible sequences. True, Thesen offers a number of "estimators" of S; but unfortunately their precision and validity are open to question.

The measure offered by Mastor is precisely one minus the measure offered by Dar-El. Consequently, if the value of one measure increases, the value of the other measure must decrease by an equal amount. We fail to see how both measures can simultaneously evaluate the increasing complexity of a given set of ANs unless they are, in fact, the one and same measure!

The measure proposed by Held, Karp and Shareshian is motivated by their proposed dynamic programming algorithm to solve the ALB problem. Counting the number of feasible sets yields the total number of memory locations required by the algorithm and supplies a rough measure of the computer time required for the calculations. Regrettably, as the authors themselves admit, the counting procedure "may prove difficult to program efficiently for a computer because of its list processing structure and demands for pattern recognition"; and that "the amount of work which this procedure requires depends on the way in which the elements are numbered".

Confining ourselves now to the measures of complexity proposed within the domain of analysis of ANs (the first column of Table 1), it is interesting to

observe that all authors were preoccupied with the problem of scheduling activities under limited availabilities of several resources. While we concede that this problem is perhaps the central problem in the study of ANs, we also note that it is the most difficult to resolve. Consequently, we conjecture that its resolution must await the realization of technological breakthroughs in the field of combinatorial optimization. This implies that, in our view, it is premature to attempt to measure the complexity of networks under limited availability of resources when both the number and quantity of resources are arbitrary.

Moreover, it seems fair to conclude from the writings of the various authors that they equate complexity to reduced parallelism among the paths of the networks, and, conversely, simplicity to parallelism (see, e.g., the last paragraph on p. 172 of ref. [23]). The rationale behind this conclusion is not evident to us. Parallelism of paths may, or may not, be a blessing depending on the objective of analysis. Indeed, if the number of feasible sequences is a correct measure of complexity, as one author suggests, then parallelism is a curse, since the number of feasible sequences of k activities in parallel is k!, the maximum possible! Furthermore, parallelism itself is not a well-defined notion, and it is of little help to refer to it as a point of reference. More interestingly, in spite of the authors' avowed respect for parallelism, or the lack of it, as the determining factor of the network's complexity, their measures are typically devoid of serious consideration of that factor. This is particularly true of the three measures clearly labeled "coefficient of network complexity (CNC)" by Pascoe, Davies and Kaimann.

Interestingly enough, the three "Experimenters" have reached the same conclusion with regard to these CNCs, namely, that there seems to be no correlation

between these coefficients and the relative prolongation of project duration due to limited availability of resources. If we interpret this conclusion correctly, it rejects the hypothesis that these CNCs reflect, in a direct or indirect manner, the impact of network structure on prolongation!

To sum up, we are rather vexed by the nonchalant manner in which the various measures have been offered. In general, there are two approaches to the construction of measures, to wit, the inductive approach and the deductive approach. The inductive approach depends on vast amounts of observation and/or experimentation, first to suggest the measure and then to verify its validity (that is, that it measures what it is supposed to measure) and its reliability (that is, the consistency of the measures obtained by different observers of the same object). On the other hand, the deductive approach relies on the characterization of a set of axioms which the sought-after measure should satisfy. This leads to either the choice of a measure from among several measures that satisfy the axioms, or to the determination that the axioms are satisfied by one and only one measure, or to the assertion that no measure exists which satisfies the stated axioms.

We regret to say that for all three coefficients of network complexity suggested thus far, there has been neither extensive observation of the phenomenon nor axiomatization of the properties of the measure. Worse still, there has not been an attempt to enumerate the factors that affect the sensation (except, of course, for the obvious values of A and N). The issue of reliability has been guaranteed to anyone who can count nodes and arcs, but there has been no attempt at validation. Except for the work of Davis and Patterson who were actually concerned with a different issue from complexity, the literature is conspicuously void of any well-designed experiment that verifies any property

of the proposed measures. (These experiments are critically evaluated in the next section; see p. 19.) In fact, the question was not even raised on whether these are ordinal or cardinal measures!

In view of all this, it seemed to us appropriate to devote the next section to an exposition of the fundamentals of measurement theory, in the hope that this may clear the air from any misunderstanding concerning what we are all attempting to do, which, in turn, may result in better measures.

2. ON MEASUREMENT AND MEASURES[†]

Measurement, in the sense of mensuration, is such an ingrained concept in our culture that one may pass through life without ever asking the question: What is "measurement" and why do we measure things? Even when the question is posed, one is usually dumbfounded for a few minutes searching for an answer, precisely because "measuring things" seems such a "natural" activity, like breathing and eating, that one undertakes it without thinking - really thinking about it. After all, one does not go through life questioning every premise or postulate on which our culture is based; one usually accepts such premises as "primitives" which require no further demonstration. Measurement seems to be such a "natural" activity.

But now that the question is raised, it demands an answer, and the most general answer of which we are aware is that given by Churchman (ref. [5], p. 84)

> "The function of measurement is to develop a method for generating a class of information that will be useful to a wide variety of problems and situations." (emphasis ours)

As Churchman himself remarked, this is a very tentative proposition that can be clarified only by appeal to the historical development and usage of measurement. The definition, however, seems to insist on two specific characteristics:

(i) that measurement is a process;

(ii) that the information generated from such a process must be useful over "a wide variety of problems and situations".

[†]This section relies heavily on the writings of Churchman [5], Hirsch [20], Leonard [30], Morgenau [32], and Stevens [43].

This latter assertion is the first explicit statement, of which we are aware, that connects measurement to the utility of the measure. This concept plays a dominant role in our argument, and we shall return to it below (see Proposition). At this juncture we find it appropriate to mention the following quotation from Morgenau (ref. [32], p. 164):

> "If observation denotes what is coercively given in sensation, that which forms the last instance of appeal in every scientific explanation or prediction, and if theory is the constructive rationale serving to understand and regularize observations, then measurement is the process that mediates between the two, the conversion of the immediate into constructs via numbers or, viewed the other way, the contact of reason with nature."

Thus, the ability of the "theoretical traveler" to both embark and debark at the "shore of empirical fact"† is the ultimate justification for concern with measurement, and the embodiment of its "use". To be sure, a sharper designation of the use of a measure will be needed in a particular application (such as the two uses of measures of network complexity mentioned in Section 1) since, interestingly enough, the use of the measure dictates the class of the measure itself and conversely, the class of the measure places certain restrictions on its use.

But the above quote from Morgenau has added one more characterization of measurement, to wit, that it is achieved <u>via numbers</u>. True, this need not be the case, since one can use the letters of the alphabet, or words, or any other class of symbols to construct the desired answer. But we shall adhere to the restriction that the process of measurement involves the <u>assignment of numbers</u>

---

†Morgenau, op. cit., p. 164.

to objects, and take that as an integral part of the definition of measurement (in consonance with the writings of Campbell [4] and Stevens [43]).

In advancing such a proposition one must be aware that numbers are, in a manner of speaking, both "good" and "bad" because of their rather exceptional properties (which derive from the mathematical theory of the field of real numbers).

Numbers are "good" because, for one thing, there are so many of them, so that one never runs out of numbers. And for another, one can use some of their properties to convey intelligence beyond the mere labeling of things; for example, houses on the right side of a street may bear odd numbers, while houses on the left side are given even numbers.

On the other hand, numbers are "bad" because of at least two reasons. First, numbers permit logical deductions that are properties of the field of the reals but not necessarily of the qualities† possessed by the members of the population about which one is discoursing. For instance, numbers are both additive and commutative; thus $4 + 5 = 5 + 4 = 9$. The "=" symbol here signifies identity: the number 4 when added to the number 5 results in a number (i.e., in another member of the same field) which is identical to the number 9; in other words the result is the number 9. But an object of temperature 1500°C (say molten steel) when "added" to another object of temperature 870°C (say molten lead) - where "addition" must be defined in a meaningful way - do not

---

†The adjective "quality" is defined precisely below. For the moment it is sufficient to apply its common usage.

result in an object of temperature 2370°C! Second, numbers possess certain properties that need not be shared by the qualities possessed by the members of the population. For example, the number 24 is not only larger than the number 12, but also it is twice 12. But the house number 24 need not be twice the size of the house number 12 unless, of course, the numbering of the houses was related, in a one-to-one basis, to their volume or floor area, which is almost never the case.

We therefore conclude that the process of measurement must be well-defined to clearly identify the permissible operations and/or interpretations relative to the resulting numbers.

Thus far we have identified measurement as a process as well as a class of entities (the numbers). To put it differently, the word "measure" connotes two usages: (i) as a verb: "to measure", which corresponds to the process of measurement, such as in "to measure the complexity of ANs"; and (ii) as a noun: "the measure", which is the number assigned to the member of the population, such as in "the measure of this table is 4 feet in length".

According to Leonard [30], the process of measurement is a three-step procedure that runs as follows:

Step A (The Recognition Step): Identify a population P of individuals which are to be measured, and a range Q of qualities such that Q is a nominal quality over P.

We elaborate. While many investigators define a quantity as a variable quality, we adhere to the definition of quantity as a set of qualities. For example, the quality "tall" varies from individual to individual, and one may legitimately view the quantity "height" as this variable quality of "tallness".

What is proposed here is to interpret the quantity "height" as a set of qualities, where each point in the set now represents a different "quality", that is, a different "tallness". The number of points in the set Q may be countably finite or denumerable, or uncountably infinite. After some reflection the reader will agree that the interpretation of quantity as a set, while admittedly novel, is nevertheless plausible and, indeed, more meaningful for the process of measurement than the alternative interpretation.

As another illustration that highlights the superiority of the set interpretation over the "variable quality" interpretation, consider the case of a paint company that offers a finite range of colors. It may offer a set of chips, with each chip bearing a different color. (The number of such chips normally does not exceed 500.) This set now defines, from the company's point of view, the quantity called "color", and one need not go beyond such identification to specify either the quantity or the qualities embodied in it. The population in this case may be identified as all painted walls of buildings in the U.S. that use this company's products, which may be a large, albeit finite, population.

Such a quantity Q is said to be nominal over P if it satisfies the following two conditions:

(i) Q is mutually exclusive over P; i.e., the subset in P holding a particular quality in Q is disjoint from any other subset in P holding any other quality in Q. (In our paint example, the walls identified with color chip "Carolina blue" should be distinct from the walls identified with the color chip "Egyptian gold".)

(ii) Q covers P; i.e., each member in P holds a quality in Q, though there may be qualities in Q not represented in P. (In the paint example, every wall in P must be identified with a color chip in Q, though some chips in Q may have never been used on any wall.)

Step B (The Identification Step): Establish operational and other criteria for determining the quality in Q possessed by a member of P.

The process of establishing the correspondence between an individual in P and a quality in Q may be simple and direct, or may be delicate and/or complicated. It may also vary with the context in which the identification is performed. Indeed, to identify the quality called "velocity" possessed by an atomic particle in an accelerator requires a process that is radically different from that required for the identification of the same quality possessed by a moving vehicle on the road. The important point to bear in mind is that the specification of this activity is an integral component of the process of measurement.

The reader will easily recognize that, so far, Steps A and B of this process may be viewed as "tagging" each member of the population P with a tag that bears a quality in the set Q. For instance: each painted wall may be given a tag with the name of a color chip; each adult American is given a tag that bears his occupation (assuming the quantity "occupations" is a nominal quantity over all adult Americans); each table is given a tag that bears its height; etc.

Step C (The Scaling Step): Establish a conventional arithmetical notation by which to record the possession of qualities in Q by members of P.

This is the activity that has traditionally occupied "measurers" since the dawn of history: the "arithmetization" of the tag, the "assignment of numbers" in Steven's definition.

A detailed exposition of this step would take us well beyond the scope of this expository note, as well as duplicate a good portion of Leonard's argument. Rather, we remark that at the termination of Step C one has a scale of measurements (denoted by s): each member of the population has been (or can be) identified as possessing one and only one quality in the set Q; that quality has been assigned a number, and the collection of these numbers is the scale of measurement.

Harking back to our discussion above about the "good" and "bad" in the use of numbers, we would be remiss if we do not alert the reader to the various properties of numbers that have been inherited by different quantities which give rise to the existence of different scales of measurement.

The minimal requirement in any process of recognition is the ability to identify equivalence. (We use the word "equivalence" in preference to the word "equality", to avoid any possible confusion (which occurs quite frequently) with the concept of "identity". Clearly, two individuals may be equivalent in occupation without being identical; i.e., without being the one and same individual.) If this is the only recognizable characteristic, then the result would be a nominal scale. The laws for constructing such a scale are simple: each quality is accorded one number such that no two qualities are accorded the same number.

Evidently, the only question that may be answered in a meaningful way under a nominal scale is the following: If member a possesses the value x in Q and member b possesses the value y in Q, is x identical to y? If yes, then a and b are equivalent relative to Q. If no, then a is not equivalent to b in Q; nothing

more can be said. Any other relation of significance must be achieved by imposing some additional rule(s) that the scale s must satisfy.

For instance if, in addition to equivalence, one is also capable of identifying relations R over Q and r over P such that R orders Q and r quasi-orders P[†] and r in P indicates R in Q, then one would be capable of constructing an ordinal scale in which the relation "<" (less than) between numbers reflects the relation R in Q and is mimicked by the relation r in P, and vice-versa. Of course, now the identification phase extends to determining the standing of any two members of the population in the relation r. This may be achieved by direct evaluation of equivalence of the two members of the population to a "standard", such as in the case of the paint company's color chips, or by direct evaluation of the two members without the intervening reference to a standard. (For instance, we may judge one object to be lighter than another by placing the two objects on the two pans of a balance scale, without knowing the weight of either.)

Evidently, an ordinal scale permits an expanded question to be posed and meaningfully answered, namely: If objects a and b are not equivalent, is arb or bra? Once again, no mathematical relation of the numbers x and y possessed by the objects a and b, respectively, is significant. For example, that y = 3x

---

[†]Note the clear distinction between the relation R that orders Q and the relation r that quasi-orders P. A binary relation R is said to order Q if it is transitive, anti-symmetric, irreflexive and, for any two members f and g of Q, either f is identically g, or fRg, or gRf. Quasi-ordering on P modifies this latter condition to read: for any two elements a and b in P, either a is equivalent to b, or arb, or bra.

or that $y - x = 4.2$ are totally without significance for the relation of a to b except insofar as these equations imply that $x < y$, whence arb.

We have dwelt at great length on the two scales, nominal and ordinal, for good reasons. First, it seems that, for a long time, the nominal scale was not recognized as a bona fide "scale". Second, the ordinal scale is a grossly misunderstood, and misused, scale. The fact that the complete ordering in Q is reflected as a quasiorder in P is a subtle point that can be highlighted only after careful preparation; which is what we tried to accomplish, perhaps inadequately. Third, from this point onwards, one encounters the more familiar scales which increase in richness of structure until one encounters the cardinal absolute scale.

The various scales and their properties are summarized in Table 2. We wish to emphasize one more entry in that Table because of its relevance to our study of the complexity of ANs.

The first scale to which one may meaningfully apply the more common statistical techniques of averaging, regression, interpolation, etc., is the cardinal interval scale. Such a scale is "invariant under linear transformation", in the sense that: (i) the difference between two points on the scale remains the same after such linear transformation, and (ii) equal differences on the scale measure equal separation between qualities.

It is significant that none of the three proposed "coefficients" of network complexity" claimed to be on a cardinal interval scale. In fact, they are not. Consequently, we are puzzled by the use of CNC-P in all three experimental investigations by Davis, Gordon and Patterson, in which linear regression was the main methodology of analysis, since such methodology is evidently inapplicable!

Scale	Basic Empirical Operations	Math. Group Structure	Typical Examples	Fundamental Properties: Arbitrary Zero	Arbitrary Unit	Ordering Intra-scale of Grades	Grade	Distance
Nominal	(a) Determination of equality	Permutation grp $x'=f(x)$ where $f(x)$ is any one-one substitution	Numbering of fertilizers Numbering of exits at airport	*[††]	*	No	*	*
Ordinal	(b) In addition to (a), Determination of R in Q and r in P s.t.r indicates R.	Isotonic grp. $x'=f(x)$ where $f(x)$ is any monotonic increasing function	Hardness of metals Student grades A,B, C,D,F Grades of cotton, leather, coffee,...	*	*	Yes	*	*
Ordered-Metric[†] (ordinal-interval)	(c) Same as ordinal but leaving interval order invariant	-ditto-	Student grades A,B, C,D,F, with such interval information as $\|A-B\|<\|B-C\|<\|C-D\|<\|D-F\|$,...	*	Yes	Yes	not proportl.	*
Linear-Interval (cardinal-interval)	(d) in addition to (b), Determination of equality of intervals or differences	linear or affine grp: $x'=ax+b$; $a>0$	Temperature (Fahrenheit or Celsius) Position Potential energy	Yes	Yes	Yes	not proportl.	proportl.
Logarithmic-Interval	(e) in addition to (b), Determination of ratios (hence equality of differences between logarithms)	Power grp: $x'=kx^n$; $k,n>0$	(Perhaps ?) prothetic continua such as subjective brightness, loudness, heaviness,...	*	Yes	Yes	proportl.	not proportl.
Ratio (Cardinal-ratio)	(f) in addition to (d), Determination of equality of ratios	Similarity grp: $x'=cx$ $c>0$	Numerosity length, density, work, time intervals,... temperature (Kelvin)	No	Yes	Yes	proportl.	proportl.
Cardinal Absolute	(g) in addition to (f), Determination of the invariant unit	none	Counting elements	No	No	Yes	proportl.	proportl.

Statistics

Scale	Measures of Location	Measures of Dispersion	Association or Correlation	Significance Tests
Nominal	Mode	Information H	Information transmitted, T Contingency Correlation	Chi square
Ordinal	Median	Percentiles	Rank-order Correlation	Sign test Run test
Ordered-Metric† (ordinal-interval)	-ditto-	-ditto-	-ditto-	-ditto-
Linear-Interval (cardinal-interval)	Arithmetic mean	Std. deviation Avge. deviation	Product-movement Correlation; Correlation ratio	t-test F-test
Logarithmic-Interval	-ditto- except applied to logarithms of scale values			
Ratio (Cardinal-ratio)	not known			
Cardinal Absolute	*			

Table 2. The Various Scales of Measurement

FOOTNOTES:

†The scale of "ordered metric" may be viewed as an aborted linear interval due to paucity of data, since it seems illogical to be able to identify the ranking of intervals according to "smaller than", but not be able to identify equality of intervals.

††An asterisk is used to denote "not applicable".

We digress momentarily. If one were to speculate on the origins of measurement, as Stevens did (ref. [43], p. 21), it seems plausible that the sensation felt by the first "measurer" was that of numerosity, to wit, that one pile of pebbles seems to possess fewer pebbles than another. To measure numerosity, the integers had to be invented; eventually leading to the discovery of the amazing correspondence between "laws" governing the natural numbers and "rules" of piles of pebbles. (For instance, the commutative law of numbers is a reflection of the observed fact that 3 pebbles added to 5 pebbles end up with a pile of 8 pebbles, the **identical** pile obtained by adding the 5 pebbles to the 3 pebbles.) In that manner, perhaps, was born the identification of mathematics with numbers (which, we have just seen, are the final outcome of measurement), to the chagrin of "blue blood" mathematicians and the delight of everyone else.

The step from "numerosity" to "length" seems a logical step, and one may still be justified in assuming that the measurement of length followed that of numerosity in short order. It is significant that both measures satisfy the axiom of additivity, which, together with other measures that obey the same axiom, led to the notion of "fundamental" measures versus "derived" measures (such as that of density). Other measures, and other classifications[†] appeared over the recorded history of scientific thought, and it would be presumptous of us to even give the appearance of writing an essay on the history of measurement in this brief note. But we do wish to emphasize the fact that while the

---

[†]For instance: extensive vs. intensive measures.

two measures alluded to above (of numerosity and length in addition to others, such as those of weight), do possess that remarkable property of additivity, there are many measures that do not possess that characteristic, such as measures of temperature, loudness, roughness and viscosity.

In fact, we wish to go one step further and propose that, in general, one cannot predict even the form that the measurement of the sensation in question will eventually assume, or the axioms such measurement will obey. As examples, we cite the measurement of "utility" (corresponding to the sensation of usefulness or worthiness) which is, in general, nonadditive, and the measurement of "information" (corresponding to the sensation of ascertaining what was previously uncertain) which is additive. Both measures rely on concepts of probability, but in radically different ways. More significantly, while all the quantities mentioned so far required one number to represent possession of a quality, the sensation of "heat" originally required two numbers to represent such possession, with the resultant vector (q,T), where q measured energy "content" and T measured heat "potential" (this was modified later to the single measure of entropy $ds = dq/T$). It is highly probable that the future will see more examples of measures that are vectors rather than single parameters. This may have the advantage of preserving the identity of several quantities, each possessing one of the "richer" scales of measurement (such as cardinal interval or ratio scales), but whose "mix" defines a quantity that can, at best, correspond to a nominal scale. In fact, we shall propose such vectors for the measurement of network complexity. We also put forward the following.

PROPOSITION: The definition of the quantity Q must also contain a statement on the ultimate use of the measure.

We readily concede that this Proposition seems absurd within the context of physical quantities; for example, the measurement of temperature should be the same independent of the use of the resultant measure. But we submit that it is not absurd at all in other contexts, such as in psychological or psychophysical measurements. Indeed, the history of survey sampling, and industrial engineering, attests to the fact that the quantity measured (such as time to perform a set of tasks) may very well yield two different scales for two different uses (in which task a is more difficult than task b in one scale, and the reverse is true in the other)! We also submit that the measurement of network complexity cannot be accomplished in a meaningful manner unless the use of the measure is specified à priori. Such definition will then constitute an integral component of the definition of the quantity under investigation.

As a consequence of our insistence on including "use" as an integral component of the quantity, the apparently "one and same" network may be "complex" on one scale and "easy" on another. We insist that the network in question is not "one and the same" network because of the difference in use of the measure.

Though we have dwelt at some length on the philosophical foundations of measurement, there are two issues of a philosophical nature that remain unanswered; they are:

(i) Is the measure of network complexity a quality possessed by the network or by the (analytical) procedure used in the analysis of the network? In other words, is "complexity" an inherent property of the network, or is it necessarily confounded by the procedure used in the analysis?

(ii) In the sequel, we propose a "calibrating function" that maps a vector of measures of disparate quantities into the real line. What properties, if any, of the original scales are "inherited" by the resulting (single) measure? For instance, if each quantity in the vector has a cardinal interval scale, would the resulting measure be also defined on a cardinal interval scale?

We postpone the discussion of these two issues until after we have delineated the proposed measures of network complexity, in the hope that our discussion will then be more meaningful.

3. SUGGESTED MEASURES OF NETWORK COMPLEXITY (MNC)

We discuss measures of network complexity under three objectives of network analysis:

1. To determine the critical path in deterministic activity networks[†] [17].
2. To determine the distribution function of the completion time of a project in probabilistic activity networks[††] [17].
3. To determine the optimal schedule under the constrained availability of a single resource.

We discuss the proposed MNC for each objective separately. For the first two objectives, we assume that activities are represented by arcs and events by nodes (the A-on-A mode of representation), while for the third objective we assume that activities are represented by nodes and events by arcs (the A-on-N mode of representation).

OBJECTIVE 1: To Determine the Critical Path

When the objective of analysis is the determination of the critical path, it seems logical to use the "number of elementary operations" as the independent variable, since the effort expended in the determination of the critical path is directly proportional to the number of such operations. These latter have two

---

[†] These are ANs in which all activity durations are assumed known deterministically.

[††] These are ANs in which some, or all, activities are random variables (hence the project completion time is also a r.v.).

components: additions and comparisons. The number of additions is equal to the sum of outgoing arcs from all nodes except the first. The number of comparisons is equal to A - N + 1, where A denotes the number of arcs and N the number of nodes in the AN.

Let e_i denote the number of outgoing arcs from node i and $E = \sum_{i \neq 1} e_i$. The desired measure of complexity is given by

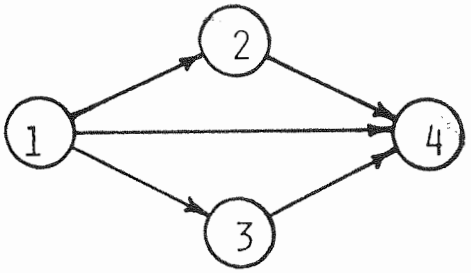
$$(MNC)_1 = g_1(E;\ A-N+1). \tag{1}$$

The function g_1 is the (monotone increasing) calibration function, which is to be determined empirically, and whose form and value depend on the procedure to calculate the critical path, the computer hardware and software, the programmer's skill, etc. For any fixed value of N, it is easy to see that $1 \leq e_i \leq N - i$, $i \neq 1$; and $N - 1 \leq A \leq N(N-1)/2$, with the upper bound on A achieved for a "completely connected network".† Thus the domain of g_1 has a well-defined origin, namely, (N-2, 0), and maximal value, namely, $(\frac{(N-1)(N-2)}{2},\ \frac{N(N-1)}{2} - N + 1)$.

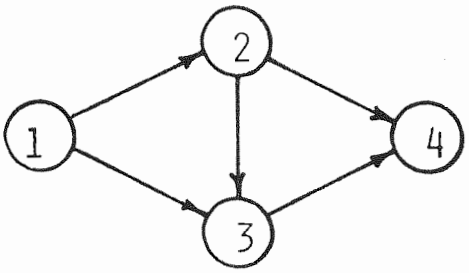
Applying the $(MNC)_1$ to the networks of Figure 1, one sees that relative to this objective networks (a) and (c) have identical measures (= $g_1(2,2)$) which is different from the measure for network (b) (= $g_1(3,2)$).

Figure 1

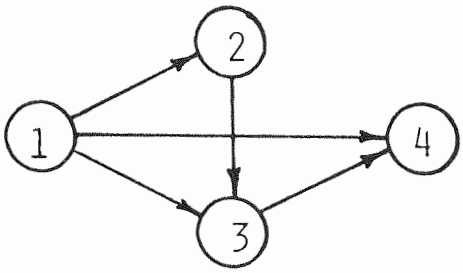
†We assume that the nodes of the network are numbered consecutively 1,2,...,N, and that an arrow leads from a small number to a larger one.



(A) $E = 2$; $A - N + 1 = 2$;
$q = 0$; $m = 0$



(B) $E = 3$; $A - N + 1 = 2$;
$q = 2$; $m = 2$



(C) $E = 2$; $A - N + 1 = 2$;
$q = 0$; $m = 3$

FIGURE 1. THREE SIMPLE NETWORKS

OBJECTIVE 2: The Analytical Determination of the Probability Distribution Function of the Terminal Event

It is assumed that the activity durations are random variables with known p.d.f.'s. We denote the time of realization of the terminal node by T_N, a random variable which is distributed as the maximum of the (finite set of) paths from the start node (see [17]).

The analytical determination of the p.d.f. of T_N involves three basic operations: the convolution of arcs in series, the multiplication of p.d.f.'s of paths in parallel, and the integration over the "conditioned" arcs (i.e., the activities whose durations were conditionally fixed in order to reduce the network to a set of paths in parallel).

The rationale for this procedure is as follows. If all paths were in parallel, the p.d.f. of the terminal event would be simply the product of the p.d.f.'s of the individual paths. For paths not in parallel, one fixes the duration of some arcs in order to reduce the paths to parallel ones. Multiplying the p.d.f.'s now yields the p.d.f. of the terminal event conditioned on the values of the specified arcs. One then removes the conditioning through multiple integration over the p.d.f.'s of these arcs. We insist on the minimum number of such arcs in order to ensure the minimum amount of effort.

The number of convolutions is equal to the number of additions in deterministic activity networks, which was determined above to be equal to $E = \sum_{i \neq 1} e_i$. The number of multiplications of p.d.f.'s is equal to the number of paths (and subpaths) in parallel which, in turn, is equal to the sum of outgoing arcs from nodes with more than one outgoing arc. Denoting this sum by m, we have that $m = (\Sigma\, e_i \mid e_i > 1)$. Finally, the minimum number of "conditioned" arcs is determined by a procedure due to Burt and Garman [3]. We denote this number by q.

Then the desired measure of network complexity is given by:

$$(MNC)_2 = g_2(q,m,E), \tag{2}$$

where, again, g_2 is the calibration function. Evidently, g_2 varies with the size of the network as given by N. However, for any fixed N, the measure varies between $g_2(0, 0, N-2)$ and $g_2(N-1, (N(N-1)/2) - 1, (N-2)(N-1)/2)$. These arguments of g_2 are easily deduced from the observation that $0 \leq q \leq N - 1$ and $0 \leq m \leq (N(N-1)/2) - 1$.

Applying this measure to the networks of Figure 1, one discovers that relative to this objective all three networks may have different measures: for (a) we have $g_2(0,0,2)$; for (b) we have $g_2(2,2,3)$; and for (c) we have $g_2(0,3,2)$.

Comparing this statement with the conclusion under Objective 1, we discover that the network in (c) is simultaneously "easy" and "complex", depending on the objective of analysis, as we previously asserted.

OBJECTIVE 3: Scheduling Tasks on a Single Resource

Scheduling precedence-related tasks under limited availability of several resources is the objective behind all the measures of network complexity proposed in the literature to date of which we are aware. However, our objective is much less ambitious: we limit the problem to resources with unit availability each that are demanded by all the activities. Our discussion will be concerned with only a single resource -- extension to several resources is straightforward.

Under this restricted objective, it seems logical to use the number of feasible sequences as the independent variable. Indeed, if the resulting n-job,

1-machine problem is NP-hard[†], this would imply that there is very little likelihood of ever finding a polynomial-bounded algorithm to solve the problem. As a result one must resort to implicit enumeration schemes that "operate" on the space of feasible sequences. Almost all problems of project scheduling under limited resources fall in this category.

On the other hand, if the resulting problem is one for which polynomial-bounded algorithms exist[††], the number of feasible sequences is no longer related to the effort expended, and the calibration of the scale should be made relative to the argument of the polynomial.

Let S denote the number of feasible sequences, then the proposed measure is given by

$$(MNC)_3 = g_3(S) \tag{3}$$

The only remaining task is to propose an algorithm by which the value of S is determined. Unfortunately, this is no minor feat since, in the presence of precedence relations, the number of feasible sequences is reduced from its

---

[†]The problem of scheduling n jobs on a single machine subject to general precedence relations has been shown to be "NP-complete" if the objective functions used are the minimization of total weighted completion time or the minimization of total tardiness (see Lawler [27] and Rinnooy Kan [37]).

[††]For certain objective functions and special cases of precedence relations polynomial-bounded algorithms have been presented. For the case of minimizing the maximum completion cost, Lawler [25] has presented an $O(n^2)$ algorithm. For the objective of minimizing the total weighted completion time, Adolphson and Hu [1] give an O(n log n) algorithm for rooted trees, and Lawler [27] presents an O(n log n) algorithm for series parallel digraphs.

maximal value of n! (which is the number of sequences in the case of n independent tasks), in a manner that is dependent on the relations themselves!

To this end, define the position matrix to be an N × N matrix $M = [m_{ij}]$ of zeros and ones, obtained from the precedence diagram as follows. The rows stand for the activities (i.e., tasks) and the columns for the possible rank order in a feasible sequence. If task i has $s \geq 0$ successors and $p \geq 0$ predecessors, then $m_{ij} = 1$ for $j = p+1, p+2, \ldots, N-s$. Let the permanent function of M, denoted by perm(M), be defined by $\Sigma m_{1i_1} \times m_{2i_2} \times \ldots \times m_{Ni_N}$ where $i_1, i_2, \ldots, i_N$ is a permutation of the numbers 1 to N, and the summation is taken over all such permutations. (Basically, the permanent of a square matrix is calculated similar to a determinant except with no sign changes.)

It is easy to demonstrate that

$$S \leq \text{perm}(M)$$

where M is the corresponding position matrix. To achieve equality - and hence get an exact count of the feasible sequences S - one must evaluate perm(M) dynamically, that is, one must modify the position matrix conditional upon the partial sequence generated at any stage of the calculation. Denoting this conditionally determined permanent by $\text{perm}(M^c)$, we assert that the desired count is given by

$$S = \text{perm}(M^c). \tag{4}$$

Appendix B gives more details on the permanent function, as well as on its dynamical updating.

Unfortunately, the determination of perm(M) is itself a procedure that experiences exponential growth in the number of tasks to be scheduled. Nijenhuis and Wilf [33] present a FORTRAN program that requires a total of $n2^{n-1}$ elementary operations (multiplication and addition); a similar computer program is given in Shriver, Eberlein and Dixon [39]. To the best of our knowledge the calculation of perm(M^c) has not been programmed for computation.

Nevertheless, the value of S may be easily determined for some special classes of networks; see Appendix B. For instance, for K parallel chains each containing n_k tasks, it is easy to see that (assuming Activity-on-Node mode of representation)

$$S = \left(\sum_{k=1}^{K} n_k\right)! \Big/ \prod_{k=1}^{K} (n_k!)$$

On the other hand, let (J,R) be an N-node network, in which J is the set of jobs and R denotes the precedence relation, that can be partitioned into m parallel but unrelated networks (J_1, R_1), (J_2, R_2),...,(J_m, R_m). Let S_i denote the number of feasible sequences for the subnetwork (J_i, R_i), and $S_{i,max}$ denote the corresponding maximum number of sequences if $R_i = \emptyset$. Then the total number of feasible sequences for the parent network (J,R) is given by

$$S = (N!) \Big/ \prod_{i=1}^{m} (S_{i,max}/S_i) \qquad (5)$$

This relation can be used recursively to determine the number of feasible sequences in more complex structures, such as in transitive series-parallel networks [28].

Applying $(MNC)_3$ of Eq. (3) to the three networks of Figure 1 (assuming one resource with unit availability) one obtains $S_a = 30$ ($\log_2 S_a = 4.91$),

$S_b = 11$ ($\log_2 S_b = 3.46$) and $S_c = 16$ ($\log_2 S_c = 4$). Consequently, for the purposes of scheduling the five activities on a single processor with unit availability,

$$(MNC)_a > (MNC)_c > (MNC)_b.$$

Interestingly, network (a) which was the easiest network for the purpose of evaluating the p.d.f. of the terminal event is the most difficult relative to scheduling the five jobs on one processor!

It would have been advantageous if the above derived measure of network complexity were still valid for the objective of scheduling a number of precedence-related activities on multiple resources with varying availabilities (the so-called resource-constrained project scheduling problem). Unfortunately, this is not the case as illustrated by the simple problem example of Fig. 2. For the precedence diagram of Fig. 2a, the parallel chain formula derived above yields S = 3. However, if this project is to be scheduled using two resources A and B with unit availability and if activities 1 and 2 require resource A and activity 3 requires resource B, the resulting number of sequences is equal to S = 6 as shown in Fig. 2c. For the problem of varying resource availabilities over time, matters get even more complicated and we do not know of an exact measure of network complexity. However, it seems evident to us that the structure of the network - in whichever way it is measured - will not be sufficient to reflect the

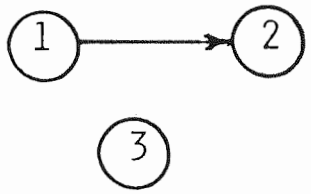
Figure 2

difficulty encountered in the resolution of such problems. In particular, the availability of resources must play an important role, and we conjecture that

the relationship between network complexity and resource availability would be as depicted in Figure 3. Indeed, if resources are only available in extremely small amounts, there will be relatively little freedom in scheduling the activities (for instance, the activities may have to be placed in series and the resulting project duration will equal the sum of activity durations), whence the complexity should be quite small (point A in Fig. 3).

Figure 3

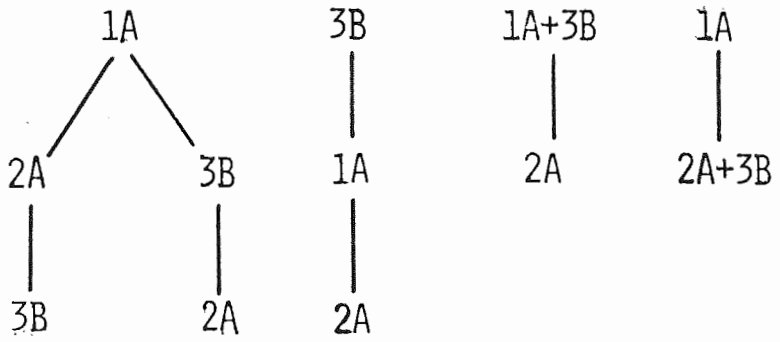
If, on the other hand, resources are amply available, the activities can simply be scheduled in parallel and the resulting project duration will be equal to the critical path length. Hence, the complexity should equal zero (point B in Fig. 3). The problem is to obtain the exact shape of the complexity curve in the region between these two extremes. Unfortunately, we do not know of any measure that is able to resolve this problem.



(A) PRECEDENCE NETWORK

Activity	Resource A	Resource B
1	1 unit	0 units
2	1 unit	0 units
3	0 units	1 unit

(B) RESOURCE REQUIREMENT TABLE



(C) FEASIBLE SEQUENCES

FIGURE 2. PROBLEM EXAMPLE FOR THE MULTI-RESOURCE CASE

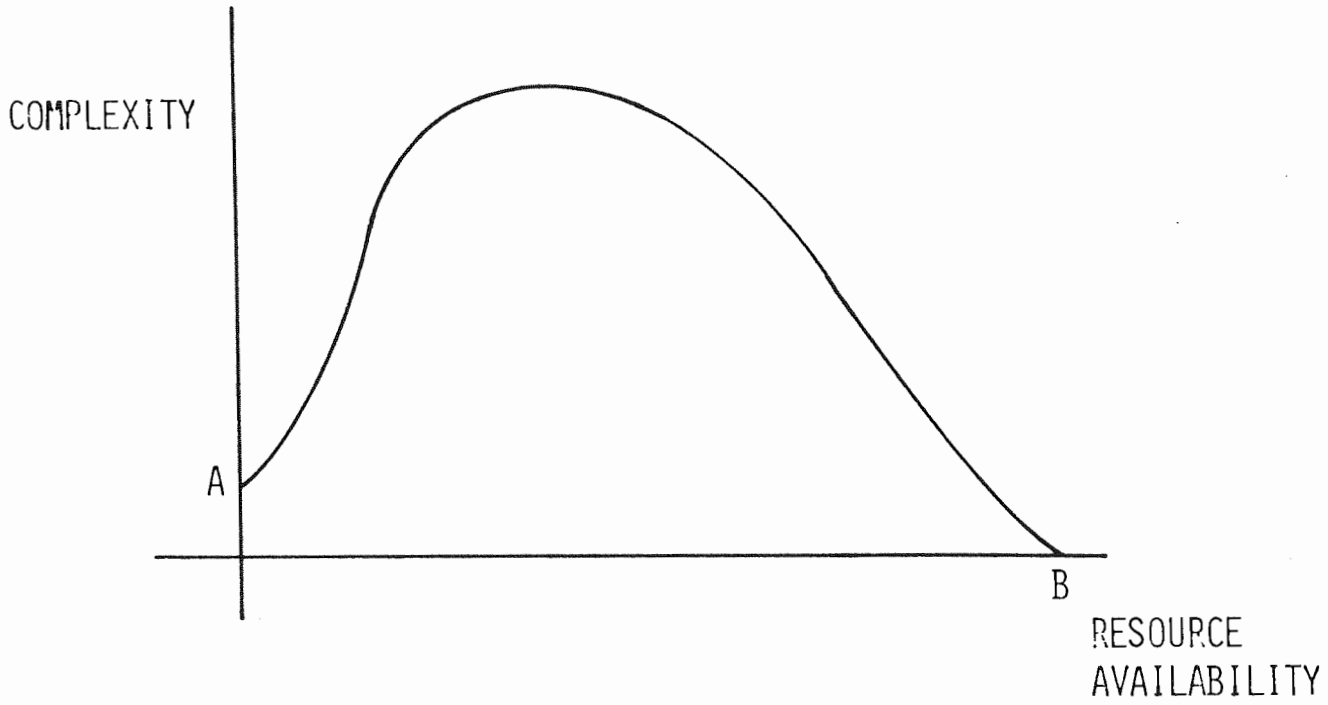


FIGURE 3. NETWORK COMPLEXITY VERSUS RESOURCE AVAILABILITY

4. SOME FURTHER CONSIDERATIONS

Our discussion demands some elaboration to clarify some of the issues that have deeper and more fundamental implications. In addition to the two points raised at the end of Section 2, we must also consider the two issues of: (iii) the axiomatic treatment of MNCs, and (iv) suggestions for further research. We treat these issues in the order of their mention in this report.

(i) The possible confounding of the MNC by the procedure of analysis

We paraphrase the issue raised on p. 24 as follows: What is the degree of validity of the proposed MNCs, especially relative to their possible confounding by the assumed (analytical) procedure used in the analysis? For instance, consider Objective 2: does $(MNC)_2$ measure an intrinsic property of the AN, or does it measure the complexity of the AN assuming that the analytical procedure used is that of "convolving" activities in series, "multiplying" activities in parallel, and "conditioning" on a subset of activities?

The issue is brought more forcefully to the fore if one considers Objective 3 and the corresponding $(MNC)_3$. The rationale of using the number of feasible sequences S as the argument of g_3 depends heavily on the assumption that an implicit enumeration scheme "of the conventional type" will be used. What if another procedure is utilized that avoids such "conventional" search; would $g_3(S)$ retain its validity?

Perhaps an analogy with similar problems in other contexts would help shed some light on the issue raised here.

Consider a linear programming problem. The number of feasible points (assuming a well-posed problem) is uncountably infinite. Prior to the develop-

ment of the (regular) simplex algorithm, the problem was considered unsolvable - that is, of infinite complexity. The simplex algorithm signalled a fundamental breakthrough in complexity because it limited the search to only finitely many points (albeit there may be a large number of them). The newly developed procedure rendered the existence of an infinite number of feasible points completely irrelevant! At the time of this writing, refinements on the algorithmic procedures of LP yield an estimate of complexity of O(mn), where m is the number of independent constraints and n is the number of variables.† Is this estimate intrinsic to the LP model, or is it dependent on the very procedure utilized in its solution?

As another example, consider the search for the maximum of a unimodal function. Typically, the width of the interval of uncertainty is taken as the measure of the complexity of the problem (say, if Fibonacci search is utilized). But if the function is analytically tractable, use of differential calculus may yield the maximum directly, and the width of the interval becomes irrelevant! Can one separate the "intrinsic" part of the problem from the procedure used to resolve it in the determination of the problem's complexity?

We concede that, indeed, the algorithm is inextricably entwined with whichever intrinsic properties we isolate and characterize as the "determining factors" under current technology (such as the number of feasible sequences S). If the technology changes then the measure may also change. The measures proposed in the previous section may be viewed in this light. They may indeed prove to be ephemeral, although we doubt that this is the case.

---

†We may also add that it is an open question whether LP problems are in class P or NP!

(ii) The calibrating function and "inherited" properties

We inquired on p. 25 about the properties of the independent parameter(s) that may be "inherited" by the dependent (or derived) measure. This inquiry is motivated by the following considerations.

At the outset, we remark that there are several instances of physical and other measurements that do "inherit" such properties. For instance, measures of both area and volume are nonlinear functions of length, but both are additive measures. Both area and volume "inherit" that property from length. More interestingly, entropy is an additive measure despite the fact that one of the two quantities defining it, namely temperature, is not additive, while the other quantity, namely the heat "content" or energy, is additive!

Second, we conjecture that measures of network complexity, such as those of concern to us here, shall continue to be determined by a vector of quantities. Each of these quantities may possess a rather rich scale (in fact, each of the quantities in the three proposed measures of Eqs. (1) to (3) possesses a cardinal absolute scale). Of course, one is always assured of a nominal scale when such a vector of quantities is available, by the simple device of associating a different number with each possible realization of measurement on the vector components. The question is: can a richer scale be assured if the individual component measures possess richer scales?

We do not know the definitive answer to this question, but we suspect that the answer is in the negative: in general, properties of individual scales are not inherited by the derived measure. (For instance, specific gravity is non-additive although both its constituent quantities, weight and volume, are additive!)

Within the context of ANs, a meaningful question is whether $(MNC)_i$; i = 1, 2 and 3; is defined on a cardinal-interval scale. We suspect that such a scale could be established; but we lack the requisite experimental evidence to demonstrate its availability.

(iii) The axiomatic treatment of MNCs

It seems natural to demand, a priori, that the proposed measures satisfy a (minimal) set of axioms ("minimal" in the sense that any other axiom may be derived from them). Although we cannot, at the present time, assert the minimality of the following two axioms, we state them as the object of future verification of minimality:

Axiom 1: Simplification of a network by removing an activity cannot lead to an increase in the MNC; and conversely, complication by adding an activity cannot decrease the MNC.

(Corollary. The calibrating functions g_i are increasing in each parameter in their arguments.)

Axiom 2: For a specific objective of analysis, equal "amounts of complication" to two networks of different complexities should augment the complexity of the more complex network by an amount not less than the augmentation in complexity of the less complex network.

Here, the "amount of complication" is itself measured along the same scale as the MNC. The axiom simply asserts that, for instance, adding one task and its

precedence relations to a complex network should increase its complexity by an amount that is at least equal to the increase in complexity of a less complex network if the same task and its precedence relations were added to it.

We assert that the three measures suggested in the previous section satisfy both axioms.

(iv) Suggestions for further research

Our discussion in the previous sections indicate the following items as avenues of future research, some of which we are currently pursuing and hope to report on in the near future.

a. Determine the functional relationship between the $(MNC)_i$ and its arguments.

b. Determine the properties of the scales of $(MNC)_i$.

c. Demonstrate the minimality of the two axioms, or else, construct the minimal set of axioms.

REFERENCES

1. Adolphson, D. and Hu, T. C., "Optimal Linear Ordering", SIAM J. Appl. Math., Vol. 25 (1973), 403-423.

2. Aho, A. V., Hopcroft, J. T. and Ullman, J. D., The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, Mass., 1974.

3. Burt, J. M. and Garman, M. B., "Conditional Monte Carlo: A Simulation Technique for Stochastic Network Analysis", Management Science, Vol. 18, 3(1971), 207-217.

4. Campbell, N. R., An Account of the Principles of Measurement and Calculation, Longmans & Green, London, 1928.

5. Churchman, C. West, "Why Measure", Chapter 4 in Ref. 6, pp. 83-94.

6. Churchman, C. West and Ratoosh, P., Measurement - Definition and Theories, Wiley & Sons, 1959.

7. Coffman, E. G., Jr., Computer and Job-Shop Scheduling Theory, Wiley, 1976.

8. Comtet, L., Advanced Combinatorics, D. Reidel Publishing Company, Dordrecht-Holland, 1974.

9. Cooper, D. F., "Heuristics for Scheduling Resource-Constrained Projects: An Experimental Investigation", Management Science, Vol. 22, 11(1976), 1186-1194.

10. Dar-El, E. M., "MALB - A Heuristic Technique for Balancing Single-Model Assembly Lines", AIIE Transactions, Vol. 5, 4(1973), 343-356.

11. _____, "Solving Large Single-Model Assembly Line Balancing Problems - A Comparative Study", AIIE Transactions, Vol. 7, 3(1975), 302-310.

12. Davies, E. M., "An Experimental Investigation of Resource Allocation in Multiactivity Projects", Operational Research Quarterly, Vol. 24, 4(1974), 587-591.

13. _____, "New Criterion for Resource Scheduling", Transportation Engineering Journal, Vol. 99, TE4(1973), 741-755.

14. Davis, E. W., "How Do You Describe a Project Network?", Paper presented at the ORSA Spring Meeting, 1972.

15. _____, "Project Summary Measures and Constrained Resource Scheduling", AIIE Transactions, Vol. 7, 2(1975), 132-142.

16. Davis, E. W. and Heidorn, G. E., "An Algorithm for Optimal Project Scheduling Under Multiple Resource Constraints", Management Science, Vol. 17, (1971), B-803-816.

17. Elmaghraby, S. E., Activity Networks: Project Planning by Network Methods, Wiley, 1977.

18. Gordon, J. H., "Project Network Categorization in an Evaluation of Heuristic Resource Allocation Techniques", unpublished Ph.D. Thesis, University of Birmingham, England, 1976.

19. Held, M., Karp, R. M. and Shareshian, R., "Assembly Line Balancing - Dynamic Programming with Precedence Constraints", Operations Research, Vol. 11, 3(1963), 442-459.

20. Hirsch, Georges, "The Notion of Characteristic Set and Its Implication for the Analysis and Development of Multicriterion Methods", Paper presented at the Conference on Multiple Criteria Decision-Making, May 21-23, 1975, Jouy-en-Josas, France.

21. Ignall, E. J., "A Review of Assembly Line Balancing", Journal of Industrial Engineering, Vol. XV, 4(1965), 244-254.

22. Johnson, T. J. R., "An Algorithm for the Resource-Constrained Project Scheduling Problem", unpublished Ph.D. Thesis, M.I.T., 1967.

23. Kaimann, R. A., "Coefficient of Network Complexity", Management Science, Vol. 21, 2(1974), 172-177.

24. _____, "Coefficient of Network Complexity: Erratum", Management Science, Vol. 21, 10(1975), 1211-1212.

25. Lawler, E. L., "Optimal Sequencing of a Single Machine Subject to Precedence Constraints", Management Science, Vol. 19, 5(1973), 544-546.

26. _____, "Graphical Algorithms and Their Complexity", Mathematical Center Tracts, 81(1976), 3-32.

27. _____, "Sequencing Jobs to Minimize Total Weighted Completion Time Subject to Precedence Constraints", to appear.

28. _____, Tarjan, R. E. and Valdez, J., "Analysis and Isomorphism of Series Parallel Digraphs", to appear.

29. Lehmer, D. H., "Permutations with Strongly Restricted Displacements", Combinatorial Theory and Its Applications - Vol. II (Erdös, P., Rényi, A. & Sos, V. T. (Eds.)), North-Holland, Amsterdam, 1970.

30. Leonard, H. S., The Use and Abuse of Measurement as a Facet of Scientific Research, University of Washington Press, Seattle, 1962.

31. Mastor, A. A., "An Experimental and Comparative Evaluation of Production Line Balancing Techniques", Management Science, Vol. 16, 11(1970), 728-746.

32. Morgenau, Henry, "Philosophical Problems Concerning the Meaning of Measurement in Physics", Chapter 8 in Ref. 6, pp. 163-176.

33. Nijenhuis, A. and Wilf, H. S., Combinatorial Algorithms, Academic Press, New York, 1975.

34. Pascoe, T. L., "Allocation of Resources - CPM", Revue Francaise de Recherche Opérationelle, No. 38 (1966), 31-38.

35. Patterson, J. H., "Project Scheduling: The Effects of Problem Structure on Heuristic Performance", Naval Research Logistics Quarterly, Vol. 23, 1(1976), 95-123.

36. Pfanzagle, J., Theory of Measurement, Wiley, 1968.

37. Rinnooy Kan, A. H. G., Machine Scheduling Problems - Classification, Complexity and Computations, Martinus Nijhoff, The Hague, 1976.

38. Riordan, J., An Introduction to Combinatorial Analysis, Wiley, 1958.

39. Shriver, B., Eberlein, P. J. and Dixon, R. D., "Algorithm 361 - Permanent Function of a Square Matrix I and II", Communications of the ACM, Vol. 12, 11(1969), 634.

40. Sidney, J. B., "Decomposition Algorithms for Single-Machine Sequencing with Precedence Relations and Deferral Costs", Operations Research, Vol. 23, 2(1975), 283-298.

41. Spencer, D. L., "An Investigation of the Relationships Between Sequencing and Scheduling Problems", unpublished Doctoral Dissertation, Texas Tech University, 1969.

42. _____, "Common Characteristics of Sequencing and Scheduling Problems", Paper presented to the Thirty-Sixth National ORSA Meeting, Miami, Florida, November, 1969.

43. Stevens, S. S., "Measurement, Psychophysics, and Utility", Chapter 2 in Ref. 6, pp. 18-63.

44. Thangavelu, S. R. and Shetty, C. M., "Assembly Line Balancing by Zero-One Integer Programming", AIIE Transactions, Vol. III (1971), 61-68.

45. Thesen, A., "Heuristic Scheduling of Activities Under Resource and Precedence Restrictions", Management Science, Vol. 23, 4(1976), 412-422.

46. _____, "Measures of the Restrictiveness of Project Networks", Networks, Vol. 7, 3(1977), 193-208.

47. Traub, J. F., Algorithms and Complexity - New Directions and Recent Results, Academic Press, 1976.

APPENDIX A

DETAILED REVIEW OF LITERATURE

1. Activity Networks Literature

Several authors suggest the so-called coefficient of network complexity (CNC) as a measure of the logic of ANs. Interestingly, given this underlying principle, the CNC has received three separate mathematical definitions, all related to the number of arcs A (activities) and the number of nodes N (events) in the network:

$$CNC - P = A/N \quad \text{(Pascoe [34])}^{\dagger}$$

$$CNC - D = 2(A-N+1)/(N-1)(N-2) \quad \text{(Davies [12, 13])}$$

$$CNC - K = A^2/N \quad \text{(Kaimann [23, 24])}$$

All three measures stem from the basic assumption that the coefficients measure the degree of "interconnectedness" of a network, where this interconnectedness (in terms of network logic) influences the time when an activity can be scheduled. In other words, for a given number of nodes an increase in the number of arcs indicates an increase in interconnectedness of the network and will correspondingly increase the value of "complexity".

---

†This measure, originated by Pascoe for activity-on-the-arrow networks, was also used in experiments performed on activity-on-node networks by Davis [14, 15] and Patterson [35]. It should be noted immediately that these authors propose several other measures for other aspects of the networks such as the shape, the size, the time and resource characteristics (see also the experiment conducted by Gordon [18]).

In addition to our general comments on the use of parallelism stated in Section 1, it seems to us that the above three measures almost ignore the concept of parallelism. To see this, consider the three simple networks shown in Figure 1: all three have A = 5 and N = 4; thus all yield the same CNC indicating equal complexity. And yet they evidently exhibit varying degrees of "parallelism". Moreover, it is well-known, and we have demonstrated in Section 3, that if the objective of analysis is to analytically determine the probability distribution of node 4, then the networks provoke different sensations of "difficulty" (and indeed consume radically different times of analysis). When the sensations (as well as empirical experience) are at variance with the indication of the measure - any of the three measures - one is forced to reject the measures as invalid!

The most coherent and most extensive computer experiments in which the above coefficients appear on the scene can be found in the work of Davis [15], Patterson [35] and Gordon [18].

Davis states the various project "summary measures" which have been proposed, such as the measures of the size, shape, logic, time and resource characteristics of networks. Then, taking these summary measures as independent variables, and the ratio (D-CP)/CP as the dependent variable, where D represents the project duration resulting from scheduling under a given set of resource availabilities using a minimum LFT (latest finishing time) heuristic, and CP represents the unconstrained duration of the critical path; he conducted an experiment on a set of 202 networks. By the application of a series of different resource constraints, this set resulted in 721 different resource-constrained problems. The result of this experiment seemed to indicate that, for the special case of constrained-resource scheduling,

summary measures based on resource requirements and availabilities appear to be more important in explaining the increase in duration beyond that of the CP than other general measures, including the complexity measure CNC-P.

Patterson [35] examines several parameters describing individual characteristics of multiple-resource constrained project scheduling problems in an attempt to predict the solution obtainable with heuristic methods for single- and multiproject data. The heuristic scheduling rules examined include least total float, greatest resource demand, greatest remaining resource demand, resource scheduling method, shortest imminent operation, greatest resource usage, earliest late finish time, most jobs possible, and random activity selection. The criterion functions investigated are: the minimization of the sum of the delays beyond the critical path length for all projects; the minimization of the sum of the total weighted delays of the projects, where the weights are determined by the size of the project measured by total resource-unit requirements (total work content) for project completion; and the minimization of the percent increase in critical path duration, where the group of projects are conjoined by dummy nodes to form one project. The parameters are divided into three categories. In one category, time and network-based parameters are computed prior to critical path analysis. This category includes, in addition to the CNC-P mentioned above, two other parameters which try to measure the interconnectedness of a network:

$$\text{T-DENSITY (Total Activity Density)}^{\dagger} = \sum_{N} \max\{0, \text{Number of Predecessor Activities} - \text{Number of Successor Activities}\}$$

$$\overline{\text{XDENSITY}} \text{ (Average Activity Density)} = \frac{\text{T-DENSITY}}{N},$$

where N denotes the nodes in the network. In the second category, time and network-based parameters are computed subsequent to critical path analysis. The third category includes resource-based parameters which are generally computed subsequent to critical path analysis. A FORTRAN program was written to calculate the parameters for two large data sets. The first data set consists of 60 multi-project scheduling problems involving 6 to 10 projects each, each project consisting of 20 to 40 activities. Thirteen different resource categories were involved, and each activity demanded fixed, positive amounts of resources from as many as 13 resource categories. The second set consists of 83 single project scheduling problems generated by Davis [15] for use in testing his bounded enumeration algorithm. The values obtained then served as independent variables in regression models to predict the heuristic performance of the above-mentioned scheduling rules, leading to the conclusion that again the resource-based parameters are the most important.

Gordon [18] conducted an experiment which was set up to test the hypothesis that a numerical categorization of project networks would form a useful

---

†The total activity density was originally proposed by Johnson [22].

basis for evaluating alternative scheduling priority rules for resource allocation procedures. To this end several regressions were run on 47 real-life networks using 14 heuristic scheduling priority rules, the dependent variable being the percentage increase of project duration above the critical path duration for three resource utilization levels. The author combined four network categorization parameters†, among which was the above-defined CNC-P, into a single scaling factor, coming to the conclusion that the individual factors, hence the CNC-P, showed little or no correlation, but that the high correlation was obtained for the scaling factor supported the above-stated hypothesis.

Thesen [45, 46] defines restrictiveness of a project scheduling network to be a measure of the degree to which the number of possible scheduling

---

†Apart from the coefficient of network complexity, the following parameters were included:

$$\text{Density} = \frac{\Sigma \text{ activity durations}}{\Sigma \text{ activity durations} + \Sigma \text{ total float}}$$

$$\text{Aspect ratio} = \frac{\text{number of activities on critical path}}{\text{max. number of activities in parallel on early start basis}}$$

$$\text{Resource factor} = \frac{\text{total requirement for resources}}{\text{max. requirement for resources on an earliest start basis}}$$

The scale factor was then computed as

$$\frac{\text{complexity}}{\text{density}} \times (\text{aspect ratio} + \text{resource factor}).$$

sequences has been restricted by the imposition of precedence requirements. He claims a useful definition for this measure to be

$$R = 1 - \log S / \log S_{max}$$

where S is the number of (precedence) feasible activity sequences for the network and S_{max} is the maximum number of sequences possible if all sequence restrictions were removed. Thus a network without precedence restrictions would have R = 0 and a fully connected project network allowing only one feasible sequence would have R = 1. For small networks, S can be determined by enumeration, and exact values for R can be obtained. For larger networks, Thesen can only obtain indirect estimations of S and R through the use of more than 40 indirect estimators, several of which are rather ambiguous in their definition and/or in the specification of their range.

2. Assembly Line Balancing Literature

In the assembly line balancing literature, several measures have been presented that seem to be related to the complexity problem dealt with in this report. Held, Karp and Shareshian [19] define a subset $S = \{J_{i_1}, J_{i_2}, \ldots, J_{i_{n(S)}}\}$ of the set of assembly jobs $J_1, J_2, \ldots, J_n$ to be feasible if $J_j \in S$ and $J_i \prec J_j$ imply that $J_i \in S$. In other words, a feasible subset is one that may be executed in some order without the prior execution of any other job. Counting the number of feasible subsets, for which they present a manual procedure, yields the total number of memory locations required for the dynamic programming algorithm and supplies a rough measure of the time required for the calculation.

Ignall [21] is the first author we are aware of who relates the number of feasible sequences to the "complexity" of the underlying precedence network. If there are r precedence relations among the N tasks, he estimates that there are roughly $N!/2^r$ distinct feasible sequences[†]. Although this measure seems to be a "good" estimator of the number of feasible sequences, it is not an exact one, since N tasks forming a single chain have exactly one possible sequence, and not $N!/2^{N-1}$ which we would otherwise obtain.

Mastor [31] asserts that the strength of the partial ordering among work elements affects the number of alternate production lines that may be established. If a partial ordering is "strong" there are few alternate ways to assign work elements to stations, while a "weak" partial ordering permits work elements to be assigned in many different ways. If k denotes the total number of ordered pairs (where the ordering is either direct or transitive) and k_{max} denotes the maximum possible number of ordered pairs (i.e., N(N-1)/2), then the order strength, OS, is defined as

$$OS = \frac{k}{k_{max}} .$$

Cooper [9] uses the order strength to characterize the structure of the project graph for generating project sets in order to test heuristics for resource-constrained project scheduling.

[†]This measure has later been used to describe the "complexity" of a set of line balancing problems (see e.g. [44]).

Dar-El [10, 11] uses the so-called flexibility ratio (F-ratio) as "an indicator of the number of feasible sequences that could be generated from an n-element assembly task". This flexibility ratio is computed from the corresponding (0-1) precedence matrix (only the portion above the main diagonal) as follows:

$$\text{F-ratio} = \frac{\text{Number of zeros in the half matrix}}{\text{Total number of cells in the partial matrix}}$$

$$= \frac{2H}{N(N-1)}$$

where H denotes the number of zeros.

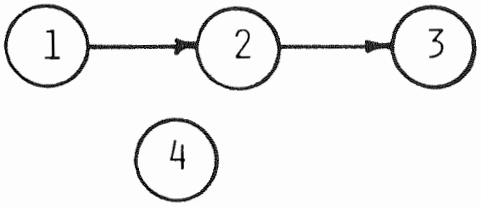
Interestingly, the order strength seems to focus on the presence of precedence relations, while the flexibility ratio focuses on the absence of precedence relations; hence both measures are in a sense complementary! To see this, take the precedence network and the corresponding half matrix of Figure A1. The total number of ordered pairs is equal to $N(N-1)/2 = k_{max} = 6$. Now the value of k used in the nominator of the OS-formula is equal to the number of 1's in the half matrix, hence k = 6; i.e., there are six ordered pairs. There are no zeros in the half matrix, hence $OS = \frac{6}{6} = 1$ and $\text{F-ratio} = \frac{0}{6} = 0$. As a result, the two measures have complementary scales:

Figure A1

for serially ordered activities, the F-ratio = 0 and OS = 1; and for jobs without precedence relations, the F-ratio = 1 and OS = 0. That neither the

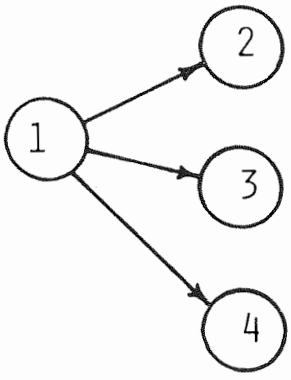
	1	2	3	4
1	1	1	1	
2		1	1	
3			1	
4				

FIGURE A1. PRECEDENCE NETWORK AND CORRESPONDING HALF MATRIX



	1	2	3	4
1	1	1	0	
2		1	0	
3			0	
4				

FIGURE A2. PRECEDENCE NETWORK AND CORRESPONDING HALF MATRIX



	1	2	3	4
1	1	1	1	
2		0	0	
3			0	
4				

FIGURE A3. PRECEDENCE NETWORK AND CORRESPONDING HALF MATRIX

F-ratio, nor the OS are valid estimators of the number of feasible sequences can be seen from the following simple example.

Computing these measures for the network of Figure A2 yields an $(\text{F-ratio})_1 = 2(3)/4(4-1) = 0.5$ and an $(\text{OS})_1 = 3/6 = 0.5$.

Figure A2

Consider now the network depicted in Figure A3. The F-ratio for this network is $(\text{F-ratio})_2 = 2(3)/4(4-1) = 0.5$ and the order strength is $(\text{OS})_2 = 3/6 = 0.5$.

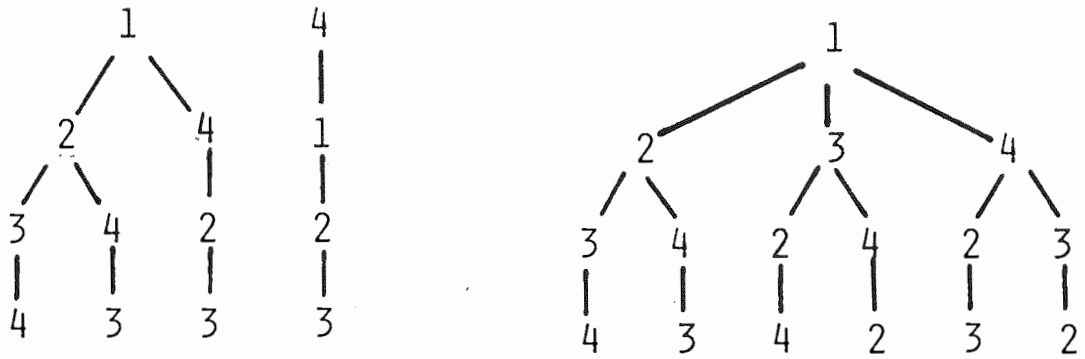
Figure A3

The number of feasible sequences, however, equals S = 4 for the network of Figure A2 and S = 6 for the network of Figure A3, as shown in Figure A4. So, although the number of feasible sequences has increased, the F-ratio and the OS remain unchanged!

Figure A4

3. Job Shop Scheduling Literature

Spencer [41, 42] recognizes the possibility of interdependence among jobs not present in "pure" job shop or flow shop sequencing models. This interdependence may be represented by technological precedence arrows inserted between previously unrelated paths of the sequencing network. The maximum number of



S = 4 (FOR FIG. A2)

S = 6 (FOR FIG. A3)

FIGURE A4. NUMBER OF FEASIBLE SEQUENCES FOR THE NETWORKS OF FIGURE A2 AND FIGURE A3.

machine schedules for this so-called dependent shop model depends upon the degree of technological dependence specified by the network diagram. To illustrate this, Spencer [42] considers a network with n activities. Each of these activities is identified by the facility required to perform the processing, say j, where j = 1,2,...,m. Let SET(j) represent the nodes requiring some facility j and let n(j) represent the total number of activities requiring facility j so that $n = \sum_{j=1}^{m} n(j)$. Now, consider the possible dominance of some node $N(i)_j$ over another node $N(k)_j$, both of which require processing on facility j. Spencer claims that this precedence, or "activity dominance", has the effect of reducing the maximum number of machine schedules possible. The expression can be written as

$$\text{MAXPOS(DS)} = \prod_{j=1}^{m} \frac{n(j)!}{\text{DENOM}(j)}$$

where

$$\text{DENOM}(j) = \prod_{i=1}^{n(j)} \left[\begin{array}{l} 1 + \text{number of activities requiring facility } j \\ \text{which are dominated by node } i, \text{ where } i,j \in \text{SET}(j) \end{array} \right]$$

and MAXPOS(DS) denotes the maximum possible number of schedules for the dependent shop. This measure is obviously not exact, as shown by the example of Figure A1. The four activities form a single chain to be processed on one machine. This chain has exactly one sequence and not

$\frac{4!}{[1+3] + [1+2] + [1+1] + [1+0]} = 2.4$, which one would obtain using MAXPOS(DS).

APPENDIX B

COUNTING THE NUMBER OF FEASIBLE SEQUENCES

1. The Permanent Function and Its Dynamical Updating

It was mentioned in Section 3 that $S \leq \text{perm}(M)$. Equality in this expression is always achieved for (a) unordered tasks, (b) single chains, and (c) the case where only three precedence-related tasks need to be scheduled. For case (a), the position matrix consists of all ones. As a result (see [29]), $\text{perm}(M) = n!$, which yields the exact number of feasible sequences. For case (b), M is an identity matrix, for which $\text{perm}(M) = 1$, which again yields the exact S-value. If three precedence-related tasks need to be scheduled, direct computation shows that $S = \text{perm}(M)$ for each meaningful case.†

In general, however, the exact count of the feasible sequences is given by $S = \text{perm}(M^C)$. The initial position matrix M can indeed be looked at as an nxn matrix of zeros and ones, in which

$$m_{ij} = \begin{cases} 1, \text{ if task } i \text{ is permitted to occupy the } j\text{th place,} \\ 0, \text{ otherwise.} \end{cases}$$

---

†We need only distinguish between the case where there is a single arc in the activity-on-node network, and the case where the number of arcs equals two. The case of three arcs need not be considered, since one of the arcs would denote a redundant (i.e., transitive) precedence relation.

From the definition of the permanent function [8, 38], it follows that perm(M) specifies the class of permutations in which the positions of the tasks after permutation are restricted. Consequently, decomposition of the original n×n position matrix (where n > 3) in order to compute the number of possible sequences recursively as the sum of the permutations in the "residual network" when a feasible task is placed ahead in sequence, might overestimate the exact number of feasible sequences. As a result, one must dynamically evaluate the permanent of M, modifying the position matrix, if necessary. This can best be illustrated using a problem example.

Direct computation of perm(M) for the precedence network of Figure B1, yields

$$\text{perm}(M) = 1\times\text{perm}\begin{pmatrix}1&1&1\\1&1&1\\1&1&1\end{pmatrix} + 1\times\text{perm}\begin{pmatrix}1&1&0\\1&1&1\\1&1&1\end{pmatrix} + 1\times\text{perm}\begin{pmatrix}1&1&0\\1&1&1\\1&1&1\end{pmatrix}$$

$$= 14,$$

which overestimates the exact number of feasible sequences.

Figure B1

However,

$$\text{perm}(M^C) = 1\times\text{perm}\begin{pmatrix}1&1&1\\1&1&1\\1&1&1\end{pmatrix} + 1\times\text{perm}\begin{pmatrix}1&1&0\\0&1&1\\1&1&1\end{pmatrix} + 1\times\text{perm}\begin{pmatrix}1&1&0\\0&1&1\\1&1&1\end{pmatrix}$$

$$= 12,$$

yields the exact S-value, since it reflects the condition that task 2 still has task 1 as a predecessor if task 3 is placed first, and task 2 has task 1 as a predecessor if task 4 if placed first.

Unfortunately, the determination of S in the above manner is a procedure which grows exponentially in the number of tasks to be scheduled. We will show below, however, that the exact value of S may be rather easily determined for some special classes of networks.

2. Counting the Number of Sequences for Special Classes of Networks

a. Parallel Chain Networks

Assertion B1. For K parallel chains each containing n_k tasks,

$$S = \Big(\sum_{k=1}^{K} n_k\Big)! \Big/ \prod_{k=1}^{K} n_k!$$

Proof. Eliminating the precedence constraints, the total number of sequences is obtained as $\big(\sum_{k=1}^{K} n_k\big)!$. For each of the n_k-task chains this total number has to be divided by $n_k!$. Since there are K chains in parallel, the result follows.

b. Trees and Forests

A network (J,R), where J = {1,2,...,n} denotes the set of tasks (nodes) and R denotes the precedence relation on the task set, is an initially rooted tree if it is a tree, and in addition there exists one task that has precedence over all other tasks. Task i is a parent node of the rooted tree (J,R) if i has at least two direct successors. Task i is a terminal parent node if it has no precedence over other parent nodes (see also [40]).

Consider any terminal parent node i, and let U = {j | i has precedence over j}. It is clear that (U,R) will be a parallel-chain network for which the formula of Assertion B1 will yield the exact number of feasible sequences. Consider now any other terminal parent node k and let Q = {j | k has precedence over j}. Then (Q,R) will be either a parallel-chain network or a network which can be factored into m parallel but unrelated subnetworks (Q_1,R_1), $(Q_2,R_2),\ldots,$ (Q_m,R_m) such that $\bigcup Q_s = Q$ and $Q_s \cap Q_r = \emptyset$ for $s \neq r$; R_s consists only of ordered pairs in Q_s; and each (Q_s,R_s) is either a parallel-chain or a tree. The following assertion then provides the basis for the direct computation of the number of sequences for trees.

Assertion B2. Let (J,R) be an n-node network that can be factored into m parallel but unrelated networks (J_1,R_1), $(J_2,R_2),\ldots,(J_m,R_m)$ such that $\bigcup J_i = J$ and $J_i \cap J_k = \emptyset$ for $i \neq k$. Let S_i denote the number of feasible sequences for (J_i,R_i) and $S_{i,max}$ the corresponding maximum number of sequences if $R_i = \emptyset$. Then the total number of feasible sequences for network (J,R) is given by

$$S = \frac{n!}{\prod_{i=1}^{m} \frac{S_{i,max}}{S_i}}$$

Proof. Eliminating the precedence constraints, the total number of sequences is equal to n!. For each of the m parallel unrelated networks this total number is divided by $S_{i,max}$. This assumes that each of the parallel unrelated networks would have a total number of feasible sequences equal to one. But since it is known that each of them has $S_i \geq 1$ feasible sequences, we have to

divide by $S_{i,max}/S_i$. Since there are m parallel unrelated networks, the result follows.

The recursive algorithm[†] for computing the total number of feasible sequences for trees can then be stated as follows:

1.a. Find any terminal parent node i.

b. Let $Q = \{j \mid i \text{ has precedence over } j\}$. Find the number of sequences S_i for (Q,R) using Assertion B1 or Assertion B2. Eliminate parent node i.

2. If parent node i is the root, terminate. Otherwise, return to 1.

Referring to Figure B2, node 3 is a terminal parent node for which $S_3 = 2$, by Assertion B1.[††] Eliminating node 3, node 4 is a terminal parent node for which Assertion B1 yields $S_4 = 2$. Retracing we find node 1 as a terminal parent node. By Assertion B2

$$S = \frac{7!}{\left(\frac{4!}{2}\right)\left(\frac{3!}{2}\right)} = 140.$$

Since node 1 is the root, we terminate.

[†]It should be noted that Assertion B2 allows the extension of this recursive algorithm to handle terminally rooted trees as well as combinations of trees in parallel, i.e. forests.

[††]The addition of dummy start and end nodes does not affect the number of feasible sequences. If networks are composed in series by making the end node of one network the start node of the other, the total number of sequences equals the product of the number of sequences of the individual networks.

Figure B2

c. Transitive Series Parallel Networks

Transitive series parallel digraphs [26, 27] are recursively defined as follows:

(1) A digraph consisting of a single node, e.g. $G = (\{i\}, \emptyset)$ is transitive series parallel.

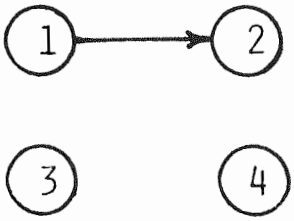
(2) If $G_1 = (N_1, A_1)$ and $G_2 = (N_2, A_2)$ where $N_1 \cap N_2 = \emptyset$ are transitive series parallel, then $G = G_1 \times G_2 = (N_1 \cup N_2, A_1 \cup A_2 \cup (N_1 \times N_2))$ is also transitive series parallel. G is said to be formed by series composition of G_1 and G_2.

(3) If $G_1 = (N_1, A_1)$ and $G_2 = (N_2, A_2)$ where $N_1 \cap N_2 = \emptyset$ are transitive series parallel then $G = G_1 \cup G_2 = (N_1 \cup N_2, A_1 \cup A_2)$ is also transitive series parallel. G is said to be formed by the parallel composition of G_1 and G_2.

(4) Only those digraphs which can be obtained by a finite number of applications of rules (1)-(3) are transitive series parallel.

A digraph G is said to be series parallel if and only if its transitive closure is transitive series parallel. Figure B3 gives an example of a series parallel digraph borrowed from Lawler [27].

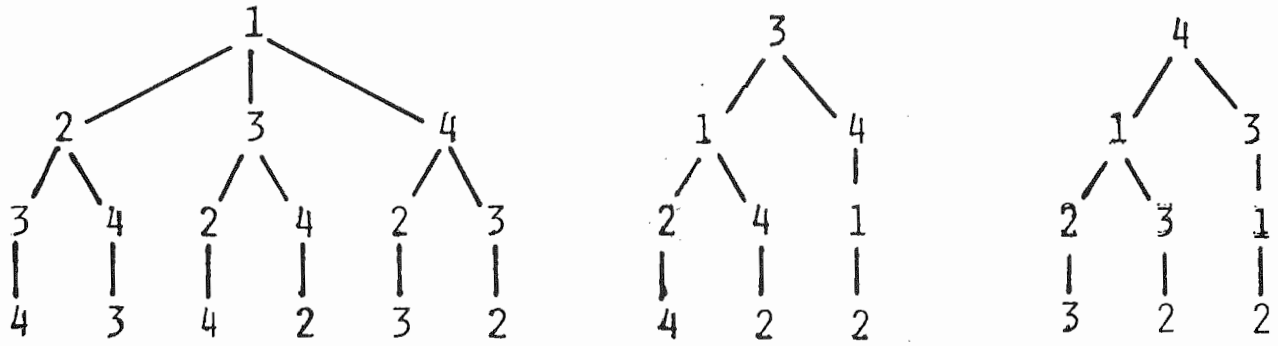
Figure B3



(A) PRECEDENCE NETWORK

$$M = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

(B) POSITION MATRIX



(C) TREES REPRESENTING S=12 FEASIBLE SEQUENCES

FIGURE B1. PROBLEM EXAMPLE FOR ILLUSTRATING THE COMPUTATION OF PERM(M^C)

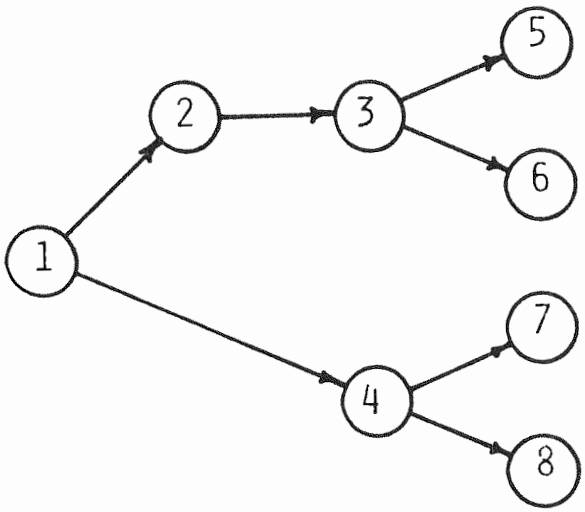


FIGURE B2. INITIALLY ROOTED TREE

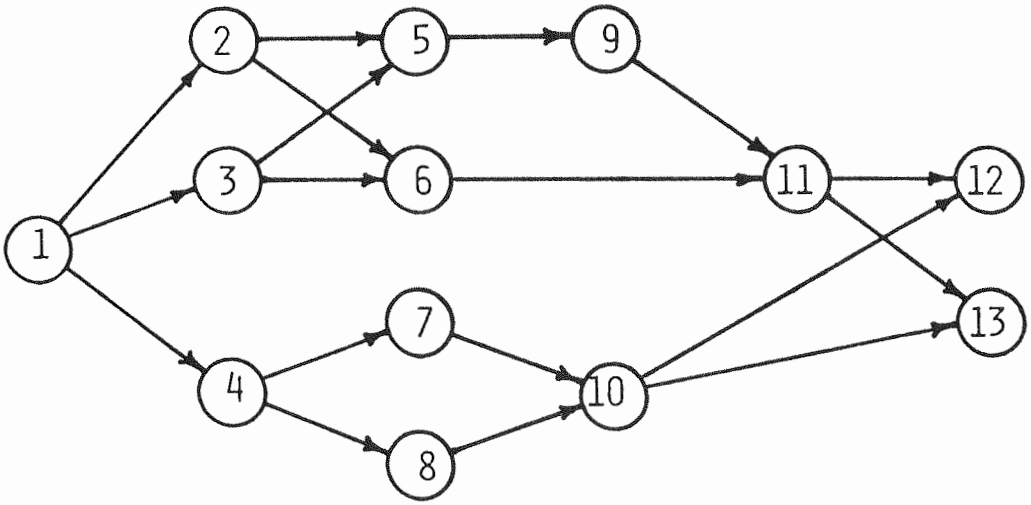


FIGURE B3. SERIES PARALLEL DIGRAPH

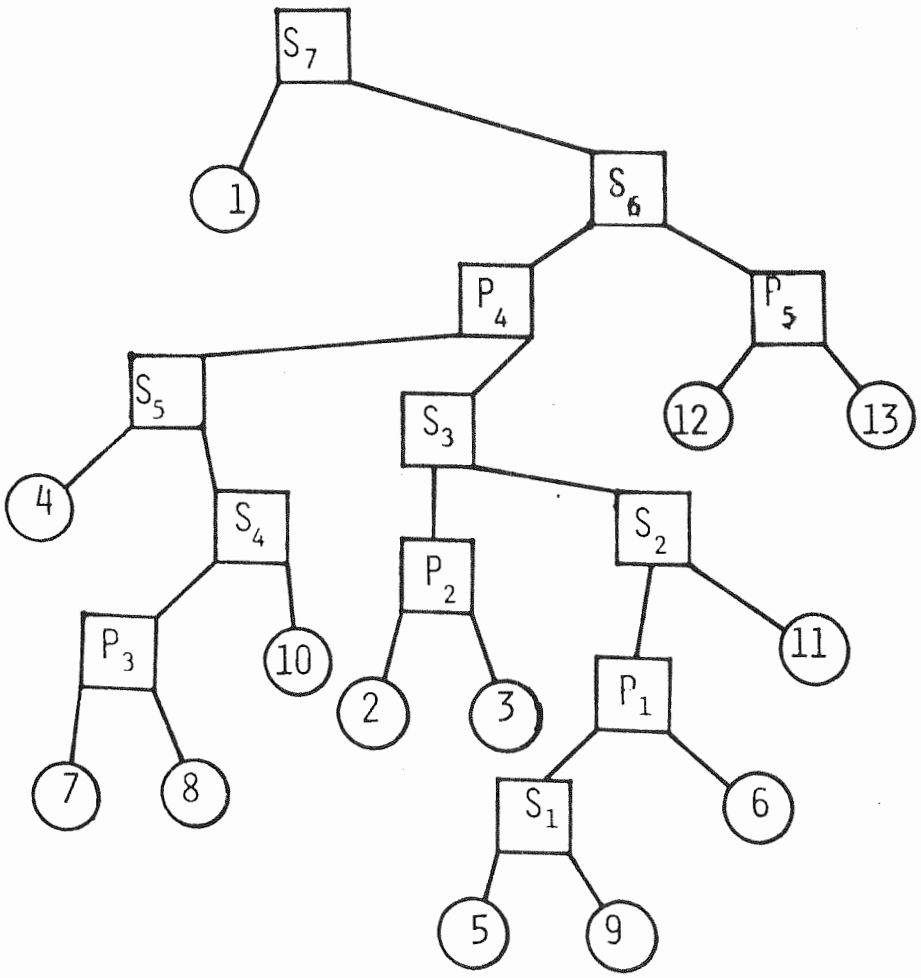


FIGURE B4. DECOMPOSITION TREE FOR THE SERIES PARALLEL DIGRAPH OF FIGURE B3.

In [26, 28] it is shown how to test a given digraph to determine if it is series parallel and, if it is, to obtain the so-called decomposition tree.† Each leaf of the decomposition tree is identified with a node of G. Each internal node marked "S" indicates the series composition of the subgraphs identified with its sons, with the convention that the left son precedes the right son. Each internal node marked "P" indicates the parallel composition of the subgraphs identified with its sons, where now the left-right ordering of sons is unimportant. The decomposition tree for the series parallel digraph of Figure B3 is given in Figure B4. The "S's" and "P's" are given subscripts to facilitate further reference.

Figure B4

The decomposition tree provides the key for the recursive computation of the number of feasible sequences for series parallel digraphs. To accomplish this, it is necessary to compute the number of feasible sequences associated with each of the internal nodes of the decomposition tree marked with "S" or "P" and retrace until the root is reached.

Internal nodes marked with "S" do not introduce any complications since the resulting number of sequences is simply obtained as the product of the

---

†The corresponding algorithm is $O(n^2)$ where n is the number of nodes in the network. An algorithm of O(m), where m is the number of arcs, will be presented in [28].

number of sequences associated with its sons. The computation of the number of feasible sequences for internal nodes labeled with "P" can always be done by applying Assertions B1 and B2, since, at worst, the computations involved will be for the parallel composition of two series parallel networks for which the associated number of feasible sequences has already been computed at lower levels of the decomposition tree.

Consider the decomposition tree of Figure B4. Internal node S_1 involves the series composition of nodes 5 and 9 into a single chain yielding one sequence. Node P_1 stands for the parallel composition of chain 5-9 and node 6 yielding $3!/(2!)(1!) = 3$ sequences. Node S_2 dictates series composition with node 11, leaving the number of sequences unchanged. P_2 calls for parallel composition of nodes 2 and 3, hence the number of sequences equals 2. The series composition prescribed by node S_3 yields $(2)(3) = 6$ sequences. Node P_3 involves parallel composition of nodes 7 and 8 (2 sequences) to be serially composed (node S_4) with node 10 (still 2 sequences) and serially composed (by S_5) with node 4 leaving the number of feasible sequences unchanged to 2. Internal node P_4 calls for the parallel composition of the subnetworks associated with nodes S_5 and S_3. Application of Assertion B2 yields $10!/(6!/6).(4!/2) = 2520$ sequences. Node P_5 asks for parallel composition of nodes 12 and 13 (2 sequences) to be serially combined with the subnetwork associated with P_4 yielding $(2)(2520) = 5040$ sequences. Finally node S_7 calls for a serial composition, hence we end up with a total of 5040 sequences.